

The screenshot shows a text editor window with two tabs: 'tree.c' and 'lqra1.c'. The 'lqra1.c' tab is active, displaying the following C code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <malloc.h>
4 struct node
5 {
6     int data;
7     struct node *left;
8     struct node *right;
9 };
10 struct node *tree;
11 void create_tree(struct node *);
12 struct node *insertElement(struct node *, int);
13 void preorderTraversal(struct node *);
14 void inorderTraversal(struct node *);
15 void postorderTraversal(struct node *);
16 int main()
17 {
18     int option, val;
19     struct node *ptr;
20     create_tree(tree);
21     clrscr();
22     do
23     {
24         printf("\n *****MAIN MENU***** \n");
25         printf("\n 1. Insert Element");
26         printf("\n 2. Preorder Traversal");
27         printf("\n 3. Inorder Traversal");
28         printf("\n 4. Postorder Traversal");
29         printf("\n\n Enter your option : ");
30         scanf("%d", &option);
31         switch(option)
32         {
33             case 1:
34                 printf("\n Enter the value of the new node : ");
35                 scanf("%d", &val);
36                 tree = insertElement(tree, val);
37                 break;
38             case 2:
39                 printf("\n The elements of the tree are : \n");
40                 preorderTraversal(tree);
41                 break;
```

The status bar at the bottom indicates 'C', 'Tab Width: 8', 'Ln 2, Col 19', and 'INS'.

The screenshot shows the same text editor window, but the 'lqra1.c' tab now displays the continuation of the C code from the previous screenshot:

```
42         preorderTraversal(tree);
43         break;
44     case 3:
45         printf("\n The elements of the tree are : \n");
46         inorderTraversal(tree);
47         break;
48     case 4:
49         printf("\n The elements of the tree are : \n");
50         postorderTraversal(tree);
51         break;
52     case 5:
53         printf("exit");
54         break;
55     default:
56         printf("Invalid");
57         break;
58 }
59 }
60 while(ch!=5);
61 }
62
63 void create_tree(struct node *tree)
64 {
65     tree = NULL;
66 }
67 struct node *insertElement(struct node *tree, int val)
68 {
69     struct node *ptr, *nodeptr, *parentptr;
70     ptr = (struct node*)malloc(sizeof(struct node));
71     ptr->data = val;
72     ptr->left = NULL;
73     ptr->right = NULL;
74     if(tree==NULL)
75     {
```

The status bar at the bottom indicates 'C', 'Tab Width: 8', 'Ln 22, Col 3', and 'INS'.

Activities Text Editor Sep 20 12:28

tree.c lqra1.c Save

```
94 }
95
96 if(val<parentptr->data)
97 parentptr->left = ptr;
98 else
99 parentptr->right = ptr;
100
101 }
102 return tree;
103 }
104 void preorderTraversal(struct node *tree)
105 {
106 if(tree != NULL)
107 {
108 printf("%d\t", tree->data);
109 preorderTraversal(tree->left);
110 preorderTraversal(tree->right);
111 }
112 }
113 void inorderTraversal(struct node *tree)
114 {
115 if(tree != NULL)
116 {
117 inorderTraversal(tree->left);
118 printf("%d\t", tree->data);
119 inorderTraversal(tree->right);
120 }
121 }
122 void postorderTraversal(struct node *tree)
123 {
124 if(tree != NULL)
125 {
126 postorderTraversal(tree->left);
127 postorderTraversal(tree->right);
128 printf("%d\t", tree->data);
129 }
130 }
131
132
133
134
135
136
```

C Tab Width: 8 Ln 22, Col 3 INS